Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A multi-streaming processor comprising:

a plurality of hardware streams for streaming one or more instruction threads;

a set of functional resources coupled to said hardware streams for processing instructions from said streams;

interstream control coupled to said functional resources including a master mode wherein more than one stream is accorded master status, and the streams accorded master status may each run a master thread exercising master control over other streams not under control of another master stream;

interrupt detection logic, coupled to interrupt signals, for providing said interrupt signals to the processor;

interrupt mapping logic, for providing configurable interrupt mapping of said interrupt signals to ones of said plurality of streams such that at least one of said interrupt signals is mapped to ones of said plurality of streams; and

interrupt logic, coupled to said interrupt mapping logic, for interrupting one or more of said streams according to said configurable interrupt mapping provided by said interrupt mapping logic;

wherein through said interrupt logic, said interrupts are detected, and at the time of their detection said ones of said plurality of streams are directed to process said interrupts.

2. (previously presented) The processor of claim 1 wherein one of said plurality of interrupts may be mapped to two or more of said plurality of hardware streams.

3. (previously presented) The processor of claim 1 wherein said plurality of interrupts comprise:

internally generated interrupts;

externally generated interrupts; and

exceptions.

4. (previously presented) The processor of claim 1 wherein said plurality of interrupts comprise:

external interrupts generated by a device external to the processor;

internal interrupts intentionally generated by special instructions executed by the processor; and

exceptions caused by execution of an instruction or a hardware error.

5. (previously presented) The processor of claim 1 wherein mapping of said interrupt signals to ones of said plurality of streams is programmable.

6. (previously presented) The processor of claim 5 wherein said configurable interrupt mapping is programmed in a data store, and said interrupt logic references said data store for mapping said interrupt signals to ones of said plurality of streams.

7. (canceled)

8. (canceled)

9. (previously presented) The processor of claim 1 wherein said interrupt signals are software interrupts generated by said hardware streams.

10. (previously presented) The processor of claim 6 wherein said data store further comprises a mask for enabling and disabling execution of said interrupt signals.

11. (previously presented) The processor of claim 1 wherein, after said interrupt mapping is determined for a detected one of said interrupt signals, said one or more hardware streams are interrupted by said interrupt logic.

12. (previously presented) The processor of claim 11 wherein said one ore more hardware streams interrupted by said interrupt logic acknowledge said interrupt, and are vectored to a service routine by said interrupt logic.

13. (previously presented) The processor of claim 12 wherein two or more of said hardware streams are interrupted by one of said interrupt signals, and wherein said interrupt logic delays vectoring any of said hardware stream to said service routine until all interrupted said hardware streams acknowledge said interrupt signal.

14. (previously presented) The processor of claim 13 wherein two of said streams acknowledging said interrupt signal are vectored to different service routines by said interrupt logic.

15. (currently amended) A method for processing an interrupt in a multi-stream processor having a plurality of hardware streams, comprising:

implementing interstream control in the processor wherein a stream may exert control functions on one or more of the plurality of hardware streams;

establishing control access privileges associated with each of the plurality of hardware streams;

exercising interstream control between the plurality of hardware streams, the interstream control utilizing said step of establishing control access privileges to assure compliance of controlled streams;

detecting the interrupt and passing the detected interrupt to interrupt mapping logic;

determining, using the interrupt mapping logic, which ones of the plurality of hardware streams are to be interrupted by the interrupt; and

interrupting the ones of the plurality of hardware streams that are to be interrupted by the interrupt.

16. (previously presented) The method of claim 15 wherein the interrupt is mapped to two or more of the hardware streams.

17. (canceled)

18. (previously presented) The method of claim 15 further comprising:

mapping of the interrupt to one or more hardware streams statically at time of manufacture of the processor.

19. (previously presented) The method of claim 15 wherein mapping of the interrupt is programmable within the processor.

20. (previously presented) The method of claim 19 wherein the mapping is programmed in a data store, and the interrupt mapping logic refers to the data store for mapping the interrupt to the ones of the plurality of hardware streams.

21. (canceled)

22. (previously presented) The method of claim 15 wherein the interrupt is an external interrupt generated by a device external to the processor.

23. (previously presented) The method of claim 15 wherein the interrupt is a software interrupt generated by one of the plurality of hardware streams.

24. (previously presented) The method of claim 20 wherein the data store comprises a mask for enabling/disabling execution of the mapped interrupt.

25. (canceled)

26. (previously presented) The method of claim 15 further comprising:

vectoring the ones of the hardware streams to service routines after the hardware streams acknowledges the interrupt.

27. (previously presented) The method of claim 26 wherein the ones of the hardware streams are interrupted by the interrupt, and wherein interrupt logic delays vectoring any of the hardware streams to a service routine until all of the ones of the hardwarestreams that are mapped to the interrupt acknowledge the interrupt.

28. (previously presented) The method of claim 27 wherein the ones of the hardware streams acknowledging the interrupt are vectored to different service routines by the interrupt logic.

Claims 29-42 (canceled)